

## Appendix A: The Model (uavmodel\_01\_11\_06)

```
(clear-all)
(pm-reset)
(load-model "act2uav_6_nov.cl")

(sgp
  :er t      ;; enable randomness
  :v t      ;; verbose
  :esc t     ;; enable subsymbolic computation
)

(pm-set-params :real-time nil :show-focus t)

(chunk-type fly-aircraft state indicator current-value current-airspeed
desired-airspeed max-desired-airspeed min-desired-airspeed previous-
airspeed current-altitude desired-altitude max-desired-altitude min-
desired-altitude current-heading desired-heading drifting heading-
deviation current-vertical-speed)
(chunk-type modify control direction magnitude)
(chunk-type instrument name location value)
(chunk-type deviation-fact current-heading desired-heading deviation)

(add-dm
  (goal isa fly-aircraft current-value nil current-airspeed 70 previous-
airspeed 67 desired-airspeed 62 max-desired-airspeed 63 min-desired-
airspeed 61 current-altitude 15000 desired-altitude 15000 max-desired-
altitude 15010 min-desired-altitude 14990 current-heading 360 desired-
heading 360 current-vertical-speed 0)
  (heading-indicator ISA instrument location heading-loc name hdg-
indicator value nil)
  (heading-loc ISA visual-location screen-x 325 screen-y 200)
  (altitude-indicator ISA instrument location altitude-loc name alt-
indicator value nil)
  (altitude-loc ISA visual-location screen-x 550 screen-y 200)
  (airspeed-indicator ISA instrument location speed-loc name airspd-
indicator value nil)
  (speed-loc ISA visual-location screen-x 180 screen-y 200)
  (vertical-speed-indicator ISA instrument location VSI-loc name vert-
speed-indicator value nil)
  (VSI-loc ISA visual-location screen-x 400 screen-y 200)
  (bank-angle-indicator ISA instrument location bank-angle-loc name
bank-indicator value nil)
  (bank-angle-loc ISA visual-location screen-x 400 screen-y 400)
  (left-deviation-1 isa deviation-fact desired-heading 360 current-
heading 359 deviation 1)
  (left-deviation-2 isa deviation-fact desired-heading 360 current-
heading 358 deviation 2)
  (left-deviation-3 isa deviation-fact desired-heading 360 current-
heading 357 deviation 3)
  (left-deviation-4 isa deviation-fact desired-heading 360 current-
heading 356 deviation 4)
  (left-deviation-5 isa deviation-fact desired-heading 360 current-
heading 355 deviation 5)
```

```

(left-deviation-6 isa deviation-fact desired-heading 360 current-
heading 354 deviation 6)
(left-deviation-7 isa deviation-fact desired-heading 360 current-
heading 353 deviation 7)
(left-deviation-8 isa deviation-fact desired-heading 360 current-
heading 352 deviation 8)
(left-deviation-9 isa deviation-fact desired-heading 360 current-
heading 351 deviation 9)
(left-deviation-10 isa deviation-fact desired-heading 360 current-
heading 350 deviation 10)
(left-deviation-11 isa deviation-fact desired-heading 360 current-
heading 349 deviation 11)
(left-deviation-12 isa deviation-fact desired-heading 360 current-
heading 348 deviation 12)
(left-deviation-13 isa deviation-fact desired-heading 360 current-
heading 347 deviation 13)
(left-deviation-14 isa deviation-fact desired-heading 360 current-
heading 346 deviation 14)
(left-deviation-15 isa deviation-fact desired-heading 360 current-
heading 345 deviation 15)
(left-deviation-16 isa deviation-fact desired-heading 360 current-
heading 344 deviation 16)
(left-deviation-17 isa deviation-fact desired-heading 360 current-
heading 343 deviation 17)
(left-deviation-18 isa deviation-fact desired-heading 360 current-
heading 342 deviation 18)
(left-deviation-19 isa deviation-fact desired-heading 360 current-
heading 341 deviation 19)
(left-deviation-20 isa deviation-fact desired-heading 360 current-
heading 340 deviation 20)
)

```

;; THESE PRODUCTIONS SELECT A UNIT TASK

```

(p synch
  =goal>
  ISA fly-aircraft
  state nil
  ==>
  =goal>
  state start
  !eval! (cycle)
)

(p just-flying-choose-heading
  =goal>
  ISA fly-aircraft
  state start
  ==>
  =goal>
  state retrieve
  indicator hdg-indicator
)

(p just-flying-choose-altitude

```

```

    =goal>
    ISA fly-aircraft
    state start
    ==>
    =goal>
    state retrieve
    indicator alt-indicator
)

(p just-flying-choose-airspeed
  =goal>
  ISA fly-aircraft
  state start
  ==>
  =goal>
  state retrieve
  indicator airspd-indicator
)

;; a single production for retrieving
;; info. about an indicator (any indicator)
(p retrieve-indicator
  =goal>
  ISA fly-aircraft
  state retrieve
  indicator =indicator
  ==>
  =goal>
  state find
  +retrieval>
  ISA instrument
  name =indicator
)

(p find-indicator
  =goal>
  ISA fly-aircraft
  state find
  =retrieval>
  ISA instrument
  location =loc
  ==>
  +visual-location>
  ISA visual-location
  nearest =loc
  =goal>
  state attend)

(p attend-indicator
  =goal>
  ISA fly-aircraft
  state attend
  =visual-location>
  ISA visual-location
  =visual-state>
  ISA module-state
  modality free

```

```

==>
+visual>
ISA visual-object
screen-pos =visual-location
=goal>
state encode
)

;; The BIND is NECESSARY here for string to integer conversion
(p encode-indicator
=goal>
ISA fly-aircraft
state encode
=visual>
ISA text
value =current-value
==>
;;!output! (=current-value)
;;!bind! =integer-value (parse-integer =current-value)
!bind! =integer-value (get-value =current-value)
=goal>
state compare
current-value =integer-value
)

;; **** PRODUCTIONS FOR HEADING UNIT TASK ****
(p compare-heading-on-course
=goal>
ISA fly-aircraft
indicator hdg-indicator
state compare
current-value =current-heading
desired-heading =current-heading
==>
=goal>
state nil
current-heading =current-heading
)

(p compare-heading-drifting-right
=goal>
ISA fly-aircraft
indicator hdg-indicator
state compare
current-value =current-heading
desired-heading =desired-heading
< current-value =desired-heading
< current-value 180
==>
=goal>
current-heading =current-heading
drifting right
state deviation
)

;; NOTE: When drifting right, can use current-heading
;; as the heading deviation, as long as it is <= 30

```

```

(p get-right-heading-deviation
  =goal>
  ISA fly-aircraft
  indicator hdg-indicator
  state deviation
  drifting right
  current-heading =current-heading
  < current-heading 30
  ==>
  =goal>
  heading-deviation =current-heading
  state retrieve
  indicator bank-indicator)

;; When deviating to the left, can't use the
;; current heading as the heading deviation, but
;; rather than include a sub-model of multi-digit
;; subtraction (which we may want in the future)
;; we will make the simplifying assumption that
;; experienced operators have chunks in memory for the
;; magnitude of deviations to the left of 0

(p compare-heading-drifting-left
  =goal>
  ISA fly-aircraft
  indicator hdg-indicator
  state compare
  current-value =current-heading
  desired-heading =desired-heading
  < current-value =desired-heading
  > current-value 180
  ==>
  =goal>
  current-heading =current-heading
  drifting left
  state deviation
  +retrieval>
  ISA deviation-fact
  current-heading =current-heading
  desired-heading =desired-heading)

(p get-left-heading-deviation
  =goal>
  ISA fly-aircraft
  indicator hdg-indicator
  state deviation
  drifting left
  =retrieval>
  ISA deviation-fact
  deviation =deviation
  ==>
  =goal>
  heading-deviation =deviation
  state retrieve
  indicator bank-indicator)

```

```
;; at this point, the generic retrieve/attend/encode
;; productions fire
```

```
(p compare-bank-with-deviation-continue-left
  =goal>
  ISA fly-aircraft
  drifting right
  indicator bank-indicator
  state compare
  current-value =current-bank-angle
  heading-deviation =deviation
  <= heading-deviation =current-bank-angle
  ==>
  =goal>
  state nil)
```

```
(p compare-bank-with-deviation-more-left
  =goal>
  ISA fly-aircraft
  drifting right
  indicator bank-indicator
  state compare
  current-value =current-bank-angle
  heading-deviation =deviation
  > heading-deviation =current-bank-angle
  ==>
  =goal>
  state change-heading-left)
```

```
(p compare-bank-with-deviation-continue-right
  =goal>
  ISA fly-aircraft
  drifting left
  indicator bank-indicator
  state compare
  current-value =current-bank-angle
  heading-deviation =deviation
  <= heading-deviation =current-bank-angle
  ==>
  =goal>
  state nil)
```

```
(p compare-bank-with-deviation-more-right
  =goal>
  ISA fly-aircraft
  drifting left
  indicator bank-indicator
  state compare
  current-value =current-bank-angle
  heading-deviation =deviation
  > heading-deviation =current-bank-angle
  ==>
  =goal>
  state change-heading-right)
```

```
(p change-heading-right
  =goal>
```

```

ISA fly-aircraft
state change-heading-right
==>
!eval! (stick-action 'right 1)
=goal>
state nil)

(p change-heading-left
=goal>
ISA fly-aircraft
state change-heading-left
==>
!eval! (stick-action 'left 1)
=goal>
state nil)

;; Still need to consider adding ...
;; (1) sensitivity to rate of improvement in the heading
;; (2) anticipation of completion of heading correction
;; (3) ability to adjust-and-release the stick

;; **** PRODUCTIONS FOR ALTITUDE UNIT TASK ****

(p altitude-at-desired-nochange
=goal>
ISA fly-aircraft
indicator alt-indicator
state compare
current-value =current-altitude
max-desired-altitude =max-desired-altitude
min-desired-altitude =min-desired-altitude
<= current-value =max-desired-altitude
>= current-value =min-desired-altitude
==>
=goal>
current-altitude =current-value
state nil
)

(p retrieve-indicator-vertical-speed-max
=goal>
ISA fly-aircraft
indicator alt-indicator
state compare
max-desired-altitude =max-desired-altitude
current-value =current-altitude
> current-value =max-desired-altitude
==>
=goal>
indicator vert-speed-indicator
state retrieve
current-altitude =current-altitude
)

(p retrieve-indicator-vertical-speed-min
=goal>
ISA fly-aircraft

```

```

indicator alt-indicator
state compare
min-desired-altitude =min-desired-altitude
current-value =current-altitude
< current-value =min-desired-altitude
==>
=goal>
indicator vert-speed-indicator
state retrieve
current-altitude =current-altitude
)

;; At this point, the attend and encode productions
;; should fire.
;; And that will bring us back to the altitude
;; comparison again...

;; IF altitude high and climbing or level (VSI>=0),
;; THEN decrease altitude
(p compare-high-altitude-and-VSI-decrease
=goal>
ISA fly-aircraft
indicator vert-speed-indicator
state compare
max-desired-altitude =max-desired-altitude
> current-altitude =max-desired-altitude
current-value =vertical-speed
>= current-value 0
==>
=goal>
state decrease-altitude
current-vertical-speed =vertical-speed
)

(p compare-high-altitude-and-VSI-no-change
=goal>
ISA fly-aircraft
indicator vert-speed-indicator
state compare
max-desired-altitude =max-desired-altitude
> current-altitude =max-desired-altitude
current-value =vertical-speed
< current-value 0
==>
=goal>
state nil
current-vertical-speed =vertical-speed
)

(p compare-low-altitude-and-VSI-increase
=goal>
ISA fly-aircraft
indicator vert-speed-indicator
state compare
min-desired-altitude =min-desired-altitude
< current-altitude =min-desired-altitude
current-value =vertical-speed

```

```

    <= current-value 0
    ==>
    =goal>
    state increase-altitude
    current-vertical-speed =vertical-speed
    )

(p compare-low-altitude-and-VSI-no-change
=goal>
ISA fly-aircraft
indicator vert-speed-indicator
state compare
min-desired-altitude =min-desired-altitude
< current-altitude =min-desired-altitude
current-value =vertical-speed
> current-value 0
==>
=goal>
state nil
current-vertical-speed =vertical-speed
)

(p high-and-slow-decrease-altitude-stick
=goal>
ISA fly-aircraft
state decrease-altitude
current-altitude =current-altitude
current-vertical-speed =current-vertical-speed
desired-altitude =desired-altitude
min-desired-airspeed =min-desired-airspeed
< current-airspeed =min-desired-airspeed
==>
!eval! (stick-action-high-and-slow-decrease-altitude 'forward
=current-altitude =desired-altitude =current-vertical-speed)
=goal>
state nil)

(p high-and-fast-decrease-altitude-throttle-and-stick
=goal>
ISA fly-aircraft
state decrease-altitude
current-altitude =current-altitude
current-vertical-speed =current-vertical-speed
desired-altitude =desired-altitude
max-desired-airspeed =max-desired-airspeed
> current-airspeed =max-desired-airspeed
==>
!eval! (throttle-action-high-and-fast-decrease-altitude 'decrease
=current-altitude =desired-altitude =current-vertical-speed)
!eval! (stick-action-high-and-fast-decrease-altitude 'forward
=current-altitude =desired-altitude =current-vertical-speed)
=goal>
state nil)

(p low-and-slow-increase-altitude-throttle-and-stick
=goal>
ISA fly-aircraft

```

```

state increase-altitude
current-altitude =current-altitude
current-vertical-speed =current-vertical-speed
desired-altitude =desired-altitude
min-desired-airspeed =min-desired-airspeed
< current-airspeed =min-desired-airspeed
==>
!eval! (throttle-action-low-and-slow-increase-altitude 'increase
=current-altitude =desired-altitude =current-vertical-speed)
!eval! (stick-action-low-and-slow-increase-altitude 'backward
=current-altitude =desired-altitude =current-vertical-speed)
=goal>
state nil)

(p low-and-fast-increase-altitude-stick
=goal>
ISA fly-aircraft
state increase-altitude
current-altitude =current-altitude
current-vertical-speed =current-vertical-speed
desired-altitude =desired-altitude
max-desired-airspeed =max-desired-airspeed
> current-airspeed =max-desired-airspeed
==>
!eval! (stick-action-low-and-fast-increase-altitude 'backward
=current-altitude =desired-altitude =current-vertical-speed)
=goal>
state nil)

;; NOTE THAT WE HAVE DOUBLED THE DENOMINATORS IN THE THROTTLE ACTIONS
FOR "SPEED CORRECT" PRODUCTIONS,
;; IN ORDER TO CUT THE MAGNITUDE OF CHANGE IN HALF.
(p high-and-speed-correct-decrease-altitude-throttle-and-stick
=goal>
ISA fly-aircraft
state decrease-altitude
current-altitude =current-altitude
current-vertical-speed =current-vertical-speed
desired-altitude =desired-altitude
min-desired-airspeed =min-desired-airspeed
max-desired-airspeed =max-desired-airspeed
<= current-airspeed =max-desired-airspeed
>= current-airspeed =min-desired-airspeed
==>
!eval! (throttle-action-high-and-correct-decrease-altitude 'decrease
=current-altitude =desired-altitude =current-vertical-speed)
!eval! (stick-action-high-and-correct-decrease-altitude 'forward
=current-altitude =desired-altitude =current-vertical-speed)
=goal>
state nil)

(p low-and-speed-correct-increase-altitude-throttle-and-stick
=goal>
ISA fly-aircraft
state increase-altitude
current-altitude =current-altitude
current-vertical-speed =current-vertical-speed

```

```

desired-altitude =desired-altitude
min-desired-airspeed =min-desired-airspeed
max-desired-airspeed =max-desired-airspeed
<= current-airspeed =max-desired-airspeed
>= current-airspeed =min-desired-airspeed
==>
!eval! (throttle-action-low-and-correct-increase-altitude 'increase
=current-altitude =desired-altitude =current-vertical-speed)
!eval! (stick-action-low-and-correct-increase-altitude 'backward
=current-altitude =desired-altitude =current-vertical-speed)
=goal>
state nil)

#|
;; WHY ARE THESE PRODUCTIONS HERE?
(p do-nothing-increase-altitude
=goal>
ISA fly-aircraft
state increase-altitude
==>
=goal>
state nil)

(p do-nothing-decrease-altitude
=goal>
ISA fly-aircraft
state decrease-altitude
==>
=goal>
state nil)
|#

;; **** PRODUCTIONS FOR AIRSPEED UNIT TASK ****

; IF speed is high and increasing or steady,
; THEN change the goal to decrease airspeed
(p compare-airspeed-decrease
=goal>
ISA fly-aircraft
indicator airspd-indicator
state compare
current-value =current-airspeed
max-desired-airspeed =max-desired-airspeed
previous-airspeed =previous-airspeed
>= current-value =previous-airspeed
> current-value =max-desired-airspeed
==>
=goal>
state decrease-airspeed
current-airspeed =current-airspeed
previous-airspeed =current-airspeed
)

; IF speed is high and decreasing,
; THEN do nothing
(p compare-high-airspeed-no-change
=goal>

```

```

ISA fly-aircraft
indicator airspd-indicator
state compare
current-value =current-airspeed
max-desired-airspeed =max-desired-airspeed
previous-airspeed =previous-airspeed
< current-value =previous-airspeed
> current-value =max-desired-airspeed
==>
=goal>
state nil
current-airspeed =current-airspeed
previous-airspeed =current-airspeed
)

; IF speed is low and decreasing or steady,
; THEN change the goal to increase airspeed
(p compare-airspeed-increase
=goal>
ISA fly-aircraft
indicator airspd-indicator
state compare
current-value =current-airspeed
previous-airspeed =previous-airspeed
min-desired-airspeed =min-desired-airspeed
<= current-value =previous-airspeed
< current-value =min-desired-airspeed
==>
=goal>
state increase-airspeed
current-airspeed =current-airspeed
previous-airspeed =current-airspeed
)

; IF speed is low and increasing,
; THEN do nothing
(p compare-low-airspeed-no-change
=goal>
ISA fly-aircraft
indicator airspd-indicator
state compare
current-value =current-airspeed
min-desired-airspeed =min-desired-airspeed
previous-airspeed =previous-airspeed
> current-value =previous-airspeed
< current-value =min-desired-airspeed
==>
=goal>
state nil
current-airspeed =current-airspeed
previous-airspeed =current-airspeed
)

(p compare-airspeed-at-desired-no-change
=goal>
ISA fly-aircraft
indicator airspd-indicator

```

```

state compare
current-value =current-airspeed
max-desired-airspeed =max-desired-airspeed
min-desired-airspeed =min-desired-airspeed
<= current-value =max-desired-airspeed
>= current-value =min-desired-airspeed
==>
=goal>
state nil
current-airspeed =current-airspeed
previous-airspeed =current-airspeed
)

(p slow-and-high-increase-airspeed-stick
=goal>
ISA fly-aircraft
state increase-airspeed
current-airspeed =current-airspeed
min-desired-airspeed =min-desired-airspeed
desired-airspeed =desired-airspeed
current-altitude =current-altitude
max-desired-altitude =max-desired-altitude
> current-altitude =max-desired-altitude
==>
!output! (current-airspeed =current-airspeed)
!output! (min-desired-airspeed =min-desired-airspeed)
!output! (current altitude =current-altitude)
!output! (max-desired altitude =max-desired-altitude)
!eval! (stick-action-slow-and-high-increase-airspeed 'forward
=current-airspeed =desired-airspeed)
=goal>
state nil)

(p slow-and-low-increase-airspeed-throttle-and-stick
=goal>
ISA fly-aircraft
state increase-airspeed
min-desired-airspeed =min-desired-airspeed
current-airspeed =current-airspeed
desired-airspeed =desired-airspeed
current-altitude =current-altitude
min-desired-altitude =min-desired-altitude
< current-altitude =min-desired-altitude
==>
!output! (current airspeed =current-airspeed)
!output! (min desired airspeed =min-desired-airspeed)
!output! (current altitude =current-altitude)
!output! (min desired altitude =min-desired-altitude)
!eval! (throttle-action-slow-and-low-increase-airspeed 'increase
=current-airspeed =desired-airspeed)
!eval! (stick-action-slow-and-low-increase-airspeed 'backward
=current-airspeed =desired-airspeed)
=goal>
state nil)

(p slow-and-correct-increase-airspeed-throttle
=goal>

```

```

ISA fly-aircraft
state increase-airspeed
min-desired-airspeed =min-desired-airspeed
current-airspeed =current-airspeed
desired-airspeed =desired-airspeed
current-altitude =current-altitude
max-desired-altitude =max-desired-altitude
min-desired-altitude =min-desired-altitude
<= current-altitude =max-desired-altitude
>= current-altitude =min-desired-altitude
==>
!output! (current-airspeed =current-airspeed)
!output! (min-desired-airspeed =min-desired-airspeed)
!output! (current altitude =current-altitude)
!output! (max-desired altitude =max-desired-altitude)
!output! (min-desired altitude =min-desired-altitude)
!eval! (throttle-action-slow-and-correct-increase-airspeed 'increase
=current-airspeed =desired-airspeed)
=goal>
state nil)

```

```

(p fast-and-correct-decrease-airspeed-throttle
=goal>
ISA fly-aircraft
state decrease-airspeed
max-desired-airspeed =max-desired-airspeed
current-airspeed =current-airspeed
desired-airspeed =desired-airspeed
current-altitude =current-altitude
max-desired-altitude =max-desired-altitude
min-desired-altitude =min-desired-altitude
<= current-altitude =max-desired-altitude
>= current-altitude =min-desired-altitude
==>
!output! (current-airspeed =current-airspeed)
!output! (max-desired-airspeed =max-desired-airspeed)
!output! (current-altitude =current-altitude)
!output! (max-desired altitude =max-desired-altitude)
!output! (min-desired altitude =min-desired-altitude)
!eval! (throttle-action-fast-and-correct-decrease-airspeed 'decrease
=current-airspeed =desired-airspeed)
=goal>
state nil)

```

```

#|
(p do-nothing-increase-airspeed
=goal>
ISA fly-aircraft
state increase-airspeed
==>
=goal>
state nil)
|#

(p fast-and-low-decrease-airspeed-stick
=goal>

```

```

ISA fly-aircraft
state decrease-airspeed
max-desired-airspeed =max-desired-airspeed
current-airspeed =current-airspeed
desired-airspeed =desired-airspeed
current-altitude =current-altitude
min-desired-altitude =min-desired-altitude
< current-altitude =min-desired-altitude
==>
!output! (current-airspeed =current-airspeed)
!output! (max-desired-airspeed =max-desired-airspeed)
!output! (current altitude =current-altitude)
!output! (min desired altitude =min-desired-altitude)
!eval! (stick-action-fast-and-low-decrease-airspeed 'backward
=current-airspeed =desired-airspeed)
=goal>
state nil)

(p fast-and-high-decrease-airspeed-throttle-and-stick
=goal>
ISA fly-aircraft
state decrease-airspeed
max-desired-airspeed =max-desired-airspeed
current-airspeed =current-airspeed
desired-airspeed =desired-airspeed
current-altitude =current-altitude
max-desired-altitude =max-desired-altitude
> current-altitude =max-desired-altitude
==>
!output! (current-airspeed =current-airspeed)
!output! (max-desired-airspeed =max-desired-airspeed)
!output! (current altitude =current-altitude)
!eval! (throttle-action-fast-and-high-decrease-airspeed 'decrease
=current-airspeed =desired-airspeed)
!eval! (stick-action-fast-and-high-decrease-airspeed 'forward
=current-airspeed =desired-airspeed)
=goal>
state nil)

#|
(p do-nothing-decrease-airspeed
=goal>
ISA fly-aircraft
state decrease-airspeed
==>
=goal>
state nil)
|#

(set-all-base-levels 2)

(spp synch :effort 0.0)

;; prefer to do something rather than nothing
;; :p set to 0.50 vs. 0.75 for these do-nothing productions
;;(spp increase-airspeed-stick :effort 0.25 :p 0.75)
;;(spp increase-airspeed-throttle :effort 0.25 :p 0.75)

```

```

;;(spp do-nothing-increase-airspeed :p 0.50)

;;(spp decrease-airspeed-stick :effort 0.25 :p 0.75)
;;(spp decrease-airspeed-throttle :effort 0.25 :p 0.75)
;;(spp do-nothing-decrease-airspeed :p 0.50)

(spp high-and-slow-decrease-altitude-stick :effort 0.25 :p 0.75)
(spp high-and-fast-decrease-altitude-throttle-and-stick :effort 0.25 :p
0.75)
(spp low-and-slow-increase-altitude-throttle-and-stick :effort 0.25 :p
0.75)
(spp low-and-fast-increase-altitude-stick :effort 0.25 :p 0.75)
(spp high-and-speed-correct-decrease-altitude-throttle-and-stick
:effort 0.25 :p 0.75)
(spp low-and-speed-correct-increase-altitude-throttle-and-stick :effort
0.25 :p 0.75)

(spp slow-and-high-increase-airspeed-stick :effort 0.25 :p 0.75)
(spp slow-and-low-increase-airspeed-throttle-and-stick :effort 0.25 :p
0.75)
(spp slow-and-correct-increase-airspeed-throttle :effort 0.25 :p 0.75)
(spp fast-and-correct-decrease-airspeed-throttle :effort 0.25 :p 0.75)
(spp fast-and-low-decrease-airspeed-stick :effort 0.25 :p 0.75)
(spp fast-and-high-decrease-airspeed-throttle-and-stick :effort 0.25 :p
0.75)

;;(spp do-nothing-decrease-altitude :p 0.50)
;;(spp do-nothing-increase-altitude :p 0.50)

;(spp just-flying-choose-heading :p 0.50)

(spp change-heading-right :effort 0.25)
(spp change-heading-left :effort 0.25)

(setf *actr-enabled-p* t)

(goal-focus goal)

```

## Appendix B: The Magnitude Equations (act2uav\_6\_nov.cl)

```

(defun stick-action-high-and-slow-decrease-altitude
  (direction current-altitude desired-altitude
   current-vertical-speed)
  (setf magnitude
    (+ (round (/ (- current-altitude desired-altitude)
                  (+ 4 (random 2)))))
      (round (/ current-vertical-speed
                  (+ 40 (random 20))))))
  (stick-action direction magnitude))

(defun stick-action-high-and-fast-decrease-altitude

```

```

        (direction current-altitude desired-altitude
         current-vertical-speed)
    (setf magnitude
      (+ (round (/ (- current-altitude desired-altitude)
                    (+ 12 (random 6))))
         (round (/ current-vertical-speed
                    (+ 120 (random 60))))))
    (stick-action direction magnitude))

(defun stick-action-low-and-slow-increase-altitude
  (direction current-altitude desired-altitude
   current-vertical-speed)
  (setf magnitude
    (- (round (/ (- desired-altitude current-altitude)
                  (+ 12 (random 6))))
       (round (/ current-vertical-speed
                  (+ 120 (random 60))))))
  (stick-action direction magnitude))

(defun stick-action-low-and-fast-increase-altitude
  (direction current-altitude desired-altitude current-vertical-
   speed)
  (setf magnitude
    (- (round (/ (- desired-altitude current-altitude)
                  (+ 4 (random 2))))
       (round (/ current-vertical-speed
                  (+ 40 (random 20))))))
  (stick-action direction magnitude))

(defun stick-action-high-and-correct-decrease-altitude
  (direction current-altitude desired-altitude current-vertical-
   speed)
  (setf magnitude
    (+ (round (/ (- current-altitude desired-altitude)
                  (+ 12 (random 6))))
       (round (/ current-vertical-speed
                  (+ 120 (random 60))))))
  (stick-action direction magnitude))

(defun stick-action-low-and-correct-increase-altitude
  (direction current-altitude desired-altitude current-vertical-
   speed)
  (setf magnitude
    (- (round (/ (- desired-altitude current-altitude)
                  (+ 12 (random 6))))
       (round (/ current-vertical-speed
                  (+ 120 (random 60))))))
  (stick-action direction magnitude))

(defun stick-action-slow-and-high-increase-air-speed
  (direction current-air-speed desired-air-speed)
  (setf magnitude
    (round (/ (- desired-air-speed current-air-speed)
              (+ 2 (random 1)))))
  (stick-action direction magnitude))

(defun stick-action-slow-and-low-increase-air-speed

```

```

    (direction current-air-speed desired-air-speed)
    (setf magnitude
      (round (/ (- desired-air-speed current-air-speed)
                  (+ 6 (random 3))))))
    (stick-action direction magnitude))

(defun stick-action-fast-and-low-decrease-air-speed
  (direction current-air-speed desired-air-speed)
  (setf magnitude
    (round (/ (- current-air-speed desired-air-speed)
                (+ 2 (random 1))))))
  (stick-action direction magnitude))

(defun stick-action-fast-and-high-decrease-air-speed
  (direction current-air-speed desired-air-speed)
  (setf magnitude
    (round (/ (- current-air-speed desired-air-speed)
                (+ 6 (random 3))))))
  (stick-action direction magnitude))

;; scaling factor = 2
(defun throttle-action-high-and-fast-decrease-altitude
  (direction current-altitude desired-altitude
    current-vertical-speed)
  (setf magnitude
    (round (* 2 (+ (/ (- current-altitude desired-altitude)
                        (+ 4 (random 2)))
                    (/ current-vertical-speed
                        (+ 120 (random 60)))))))
  (throttle-action direction magnitude))

;; scaling factor = 2
(defun throttle-action-low-and-slow-increase-altitude
  (direction current-altitude desired-altitude
    current-vertical-speed)
  (setf magnitude
    (round (* 2 (- (/ (- desired-altitude current-altitude)
                        (+ 4 (random 3)))
                    (/ current-vertical-speed
                        (+ 120 (random 60)))))))
  (throttle-action direction magnitude))

(defun throttle-action-high-and-correct-decrease-altitude
  (direction current-altitude desired-altitude
    current-vertical-speed)
  (setf magnitude
    (round (+ (/ (- current-altitude desired-altitude)
                  (+ 8 (random 4)))
              (/ current-vertical-speed
                  (+ 120 (random 60))))))
  (throttle-action direction magnitude))

(defun throttle-action-low-and-correct-increase-altitude
  (direction current-altitude desired-altitude
    current-vertical-speed)
  (setf magnitude
    (round (- (/ (- desired-altitude current-altitude)
                  (+ 8 (random 4)))
              (/ current-vertical-speed
                  (+ 120 (random 60))))))
  (throttle-action direction magnitude))

```

```

        (+ 8 (random 4)))
      (/ current-vertical-speed
        (+ 120 (random 60)))))
    (throttle-action direction magnitude))

;; scaling factor = 2
(defun throttle-action-slow-and-low-increase-airspeed
  (direction current-airspeed desired-airspeed)
  (setf magnitude
    (round (* 2 (+ (- desired-airspeed current-airspeed)
                    (+ 2 (random 1))))))
  (throttle-action direction magnitude))

;; scaling factor = 1.5
(defun throttle-action-slow-and-correct-increase-airspeed
  (direction current-airspeed desired-airspeed)
  (setf magnitude
    (round (* 1.5 (+ (- desired-airspeed current-airspeed)
                    (+ 2 (random 1))))))
  (throttle-action direction magnitude))

;; scaling factor = 1.5
(defun throttle-action-fast-and-correct-decrease-airspeed
  (direction current-airspeed desired-airspeed)
  (setf magnitude
    (round (* 1.5 (+ (- current-airspeed desired-airspeed)
                    (+ 2 (random 1))))))
  (throttle-action direction magnitude))

;; scaling factor = 2
(defun throttle-action-fast-and-high-decrease-airspeed
  (direction current-airspeed desired-airspeed)
  (setf magnitude
    (round (* 2 (+ (- current-airspeed desired-airspeed)
                    (+ 2 (random 1))))))
  (throttle-action direction magnitude))

```